Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Defence,
Civil Protection and Sport DDPS
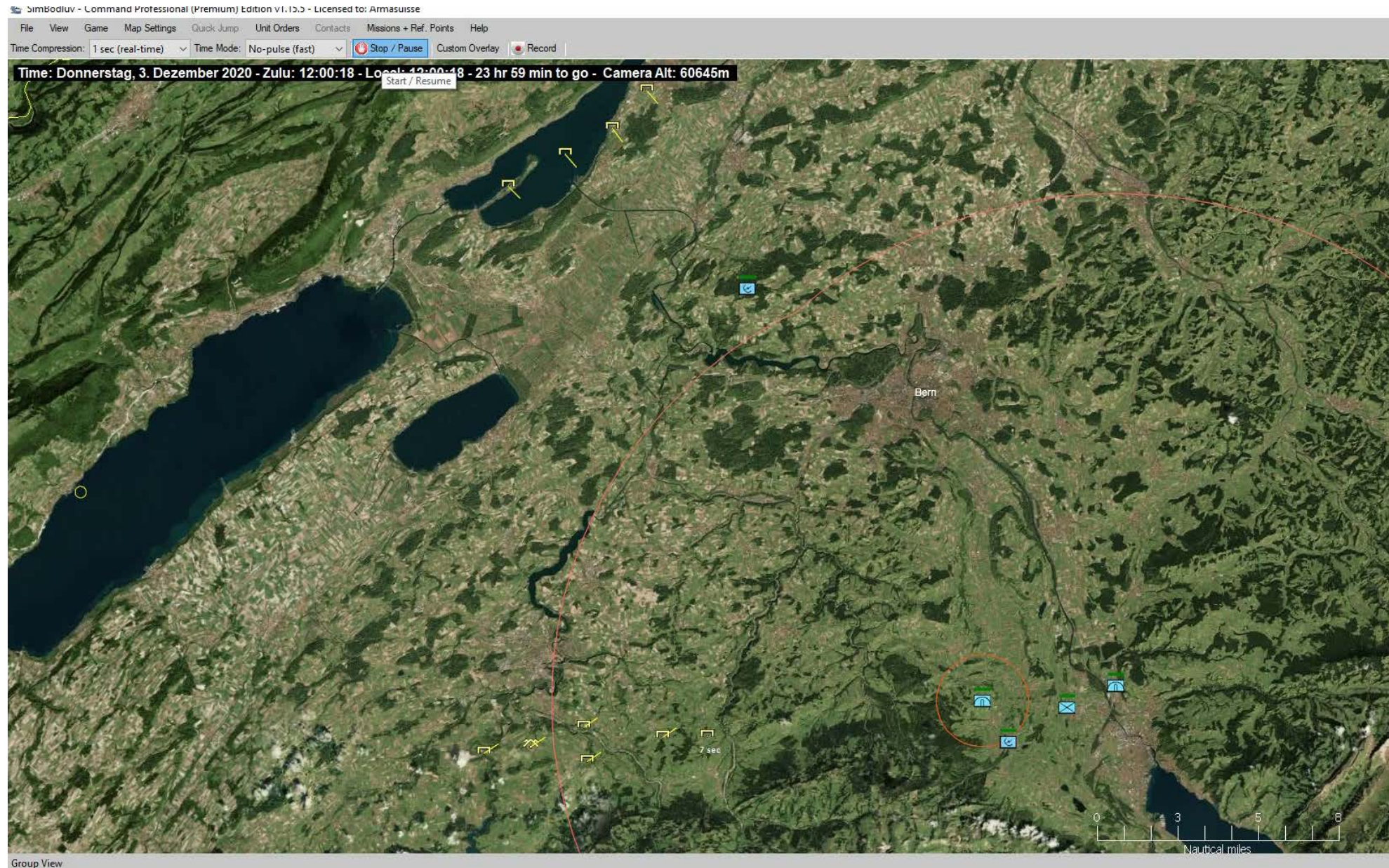**armasuisse**
Science and Technology

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization

Matthias Sommer, Oleg Szehr

Air Defense Scenario

Time: Donnerstag, 3. Dezember 2020 - Zulu: 12:00:18 - Local: 12:00:18 - 23 hr 59 min to go - Camera Alt: 60645m

SAM Plt (RBS 70 Bolide ASRAD-R x 2 [ItO 2005])
0 deg
0 kts

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr
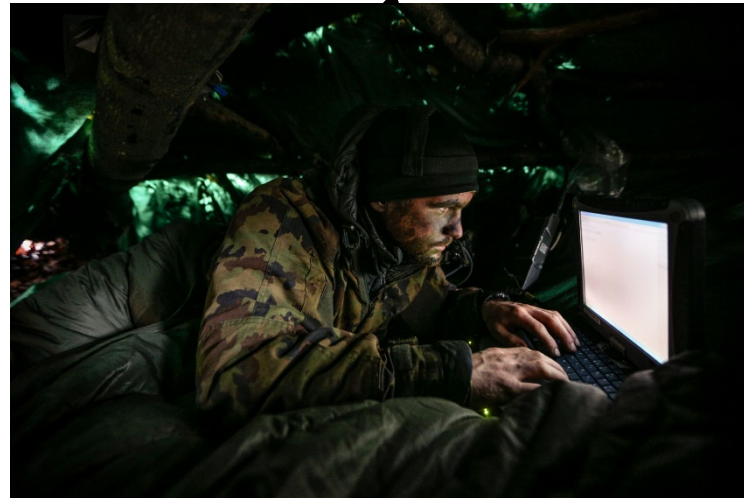
MS 36369/00

Wargaming: Human - Human

- Training tactical-strategical thinking
- Conflict analysis
- Mission planning
- Validation of known Courses of Action

- Think inside the box
- Slow
- Restricted reproducibility

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
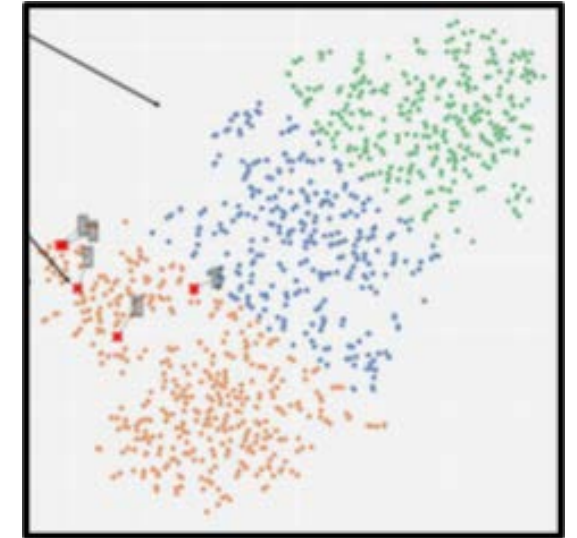DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

Red Teaming: Human - AI

- Training tactical-strategical thinking
- Conflict analysis
- Mission planning and optimization

- Diverse and new Courses of Action
  (Think outside of the box)
- Slow
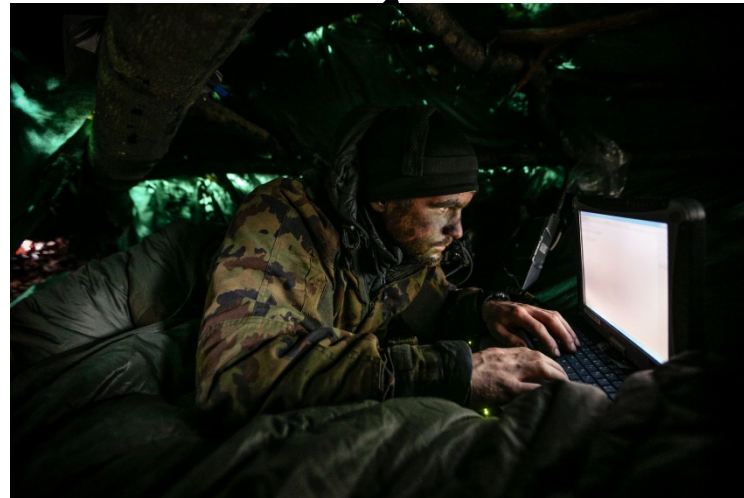
Data Farming: AI - AI

- Concept development and exerpimentation (CD&E)
- Armed Forces development
- Conflict analysis
- Procurement

- Diverse and new Courses of Action (Think outside of the box)
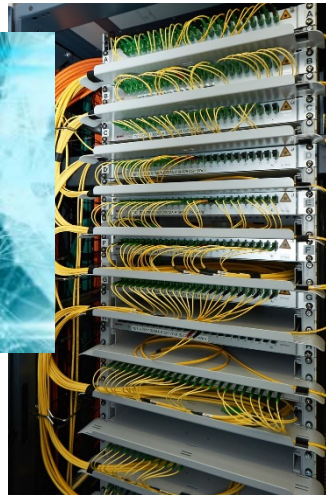- Fast
- Solution space can be scanned extensively

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

AI: Reinforcement Learning & Tree Search

- Neural networks are learning by "Trial-and-Error"


```python
class MCTS():
    #Monte Carlo Tree Search

    def __init__(self, game, nnet, args):
        self.game = game
        self.nnet = nnet
        self.args = args
        self.Qsa = {}
        self.Nsa = {}
        self.Ns = {}
        self.Ps = {}

        self.Es = {}
        self.Vs = {}

    def getActionProb(self, canonicalBoard, temp=1):
        #print('canonical ' + str(canonicalBoard))
        for i in range(self.args['numMCTSSims']):
            self.search(canonicalBoard)
        #print(self.Nsa)
```
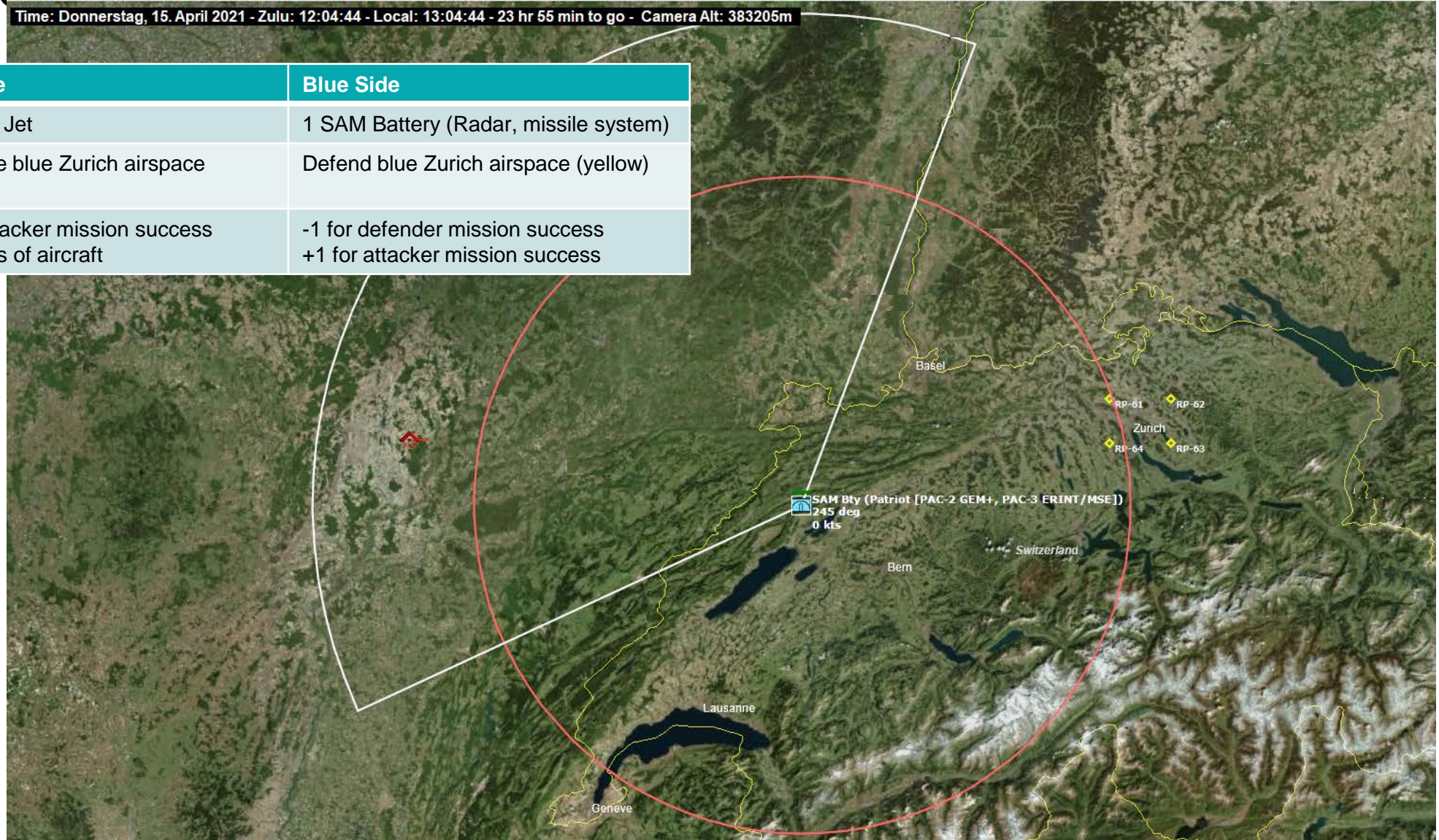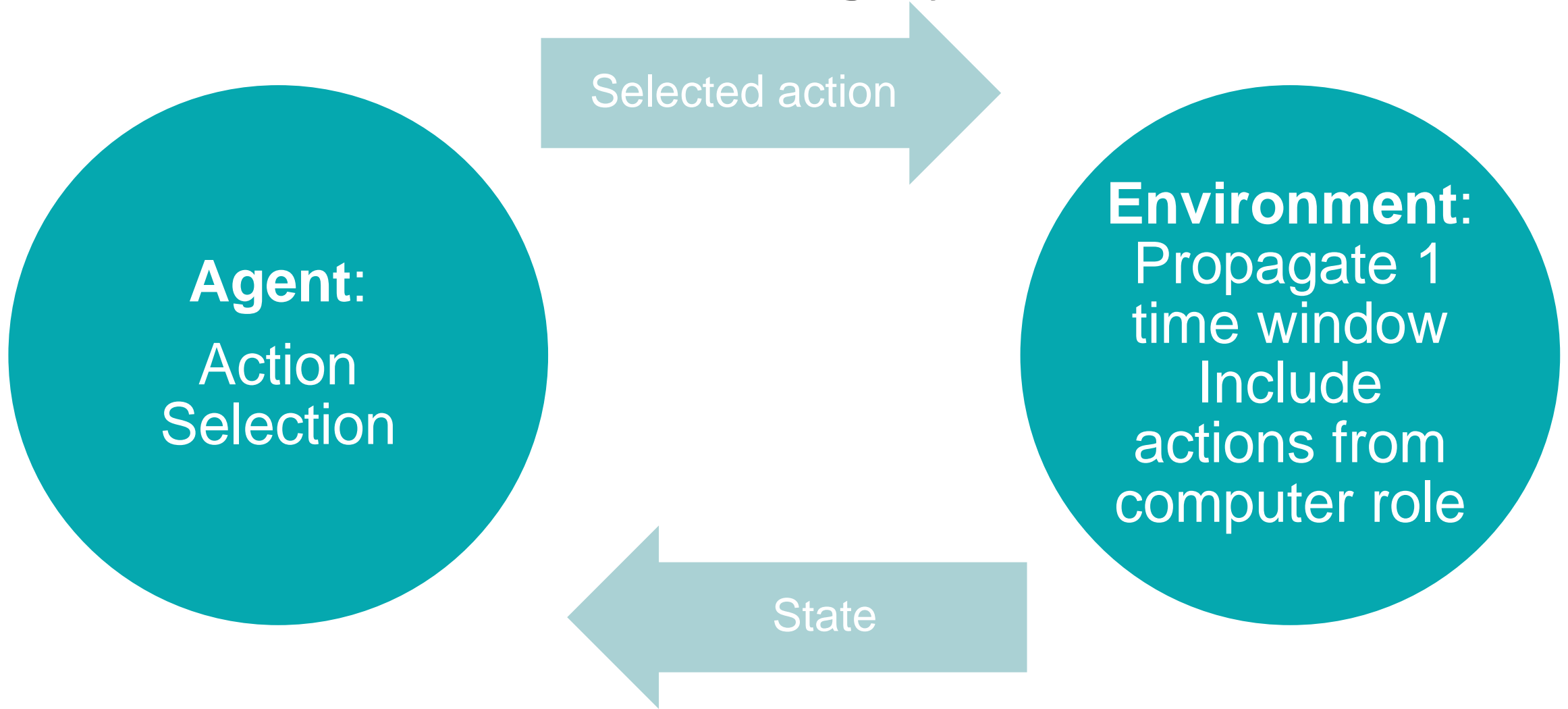
Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369 / 00

A Simplistic Air Defense Scenario

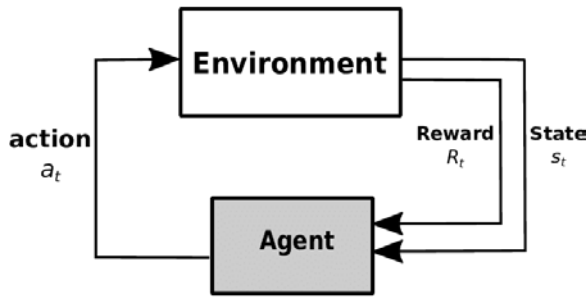|	Red Side	Blue Side
Assets	1 Fighter Jet	1 SAM Battery (Radar, missile system)
Mission	Penetrate blue Zurich airspace (yellow)	Defend blue Zurich airspace (yellow)
Red Scoring	+1 for attacker mission success -1 for loss of aircraft	-1 for defender mission success +1 for attacker mission success

Time: Donnerstag, 15. April 2021 - Zulu: 12:04:44 - Local: 13:04:44 - 23 hr 55 min to go - Camera Alt: 383205m

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

Discrete Action Scheduling Cycle

Agent:
Action Selection

Selected action

Environment:
Propagate 1 time window
Include actions from computer role

State

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
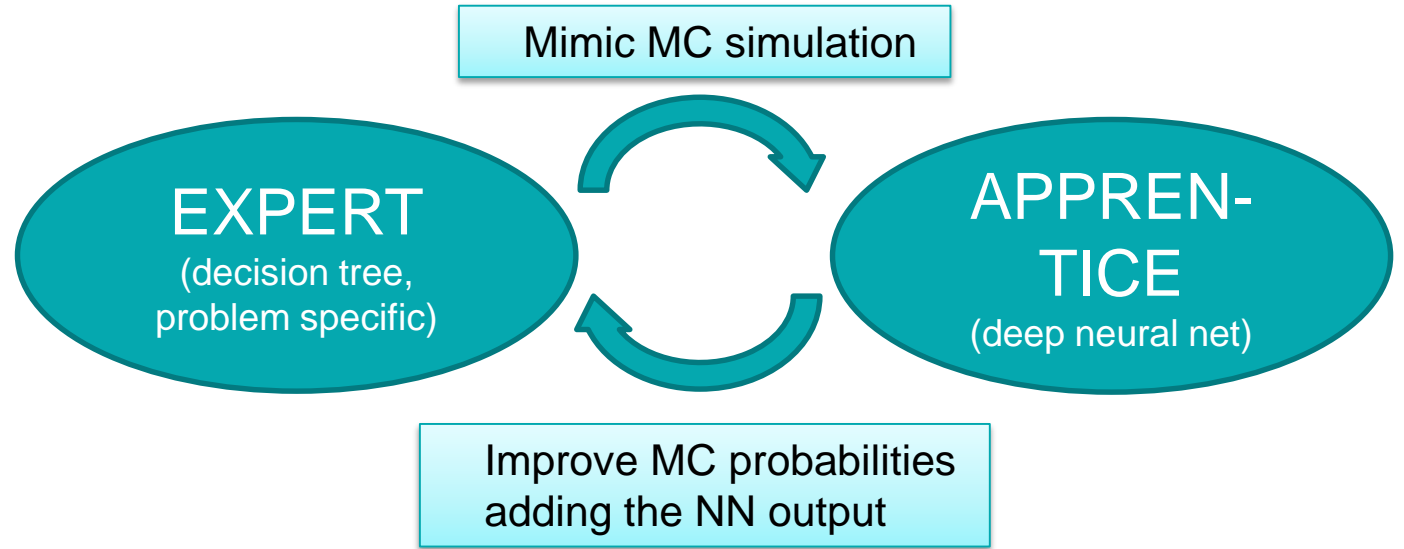DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

AI: Neural Monte Carlo Tree Search (nMCTS)

Reinforcement Learning (RL):
through trial, error and successive self-optimization

nMCTS : Two components, improve one another iteratively

Mimic MC simulation

EXPERT
(decision tree, problem specific)

APPREN-TICE
(deep neural net)

Improve MC probabilities adding the NN output

- Neural RL achieves super-human performance in Go, Chess, Hex …
- *nMCTS* combines techniques from
  o Monte Carlo (MC) simulation
  o game tree search
  o deep learning

EXPERT:
- Run MC simulation
- Record outcomes
  o update quality values of state-action pairs
  o Update MC probabilities
- Perform simulation again, update, …, ...

APPRENTICE:
- Train neural network to imitate the behavior of MC
  o Mimic action probabilities (train on tree policy targets)
 o Mimic values of states (train on rewards)

AI: Neural Monte Carlo Tree Search II

EXPERT (MCTS):

+ Provides accurate actions and valuation.

- Requires significant computational effort. Is slow.

- Evaluation of similar states requires full simulation 'from scratch'

APPRENTICE (Neural Network):

- Cannot learn actions/valuation by gradient descent in complex scenarios.

+ Once trained to imitate EXPERT: Fast access to EXPERT advice

+ Generalizes EXPERT advice to similar states.
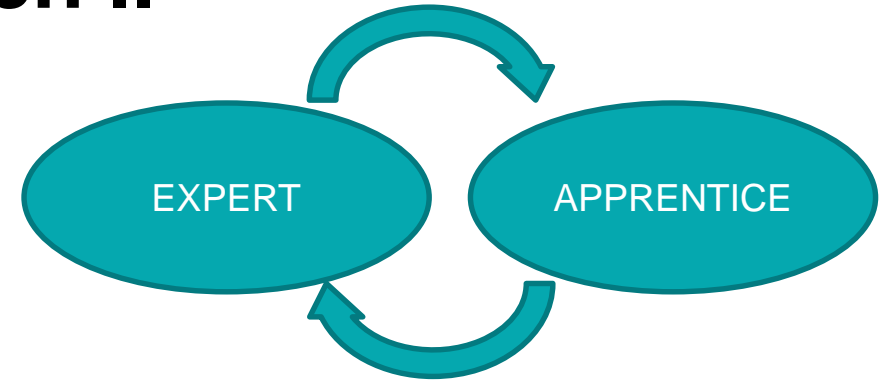
Each training iteration consists of:
- Multiple neural network-guided MC simulations
- Re-training of the neural network and update of MC probabilities
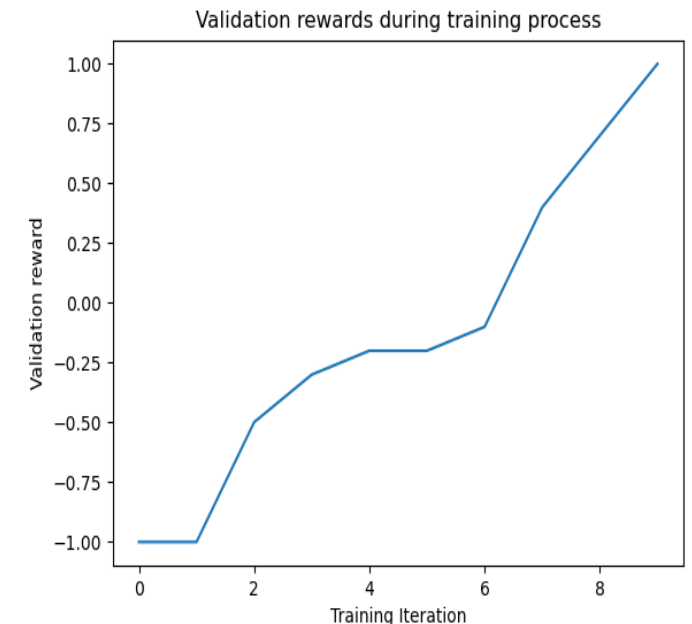- Validation to ensure improvement

EXPERT   APPRENTICE

Validation rewards during training process

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
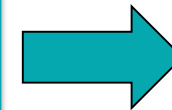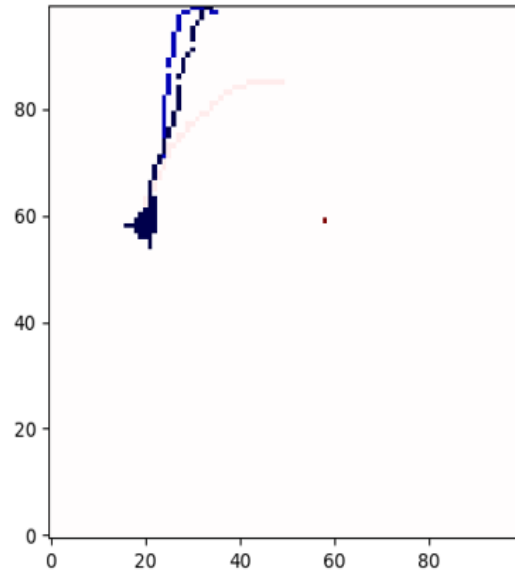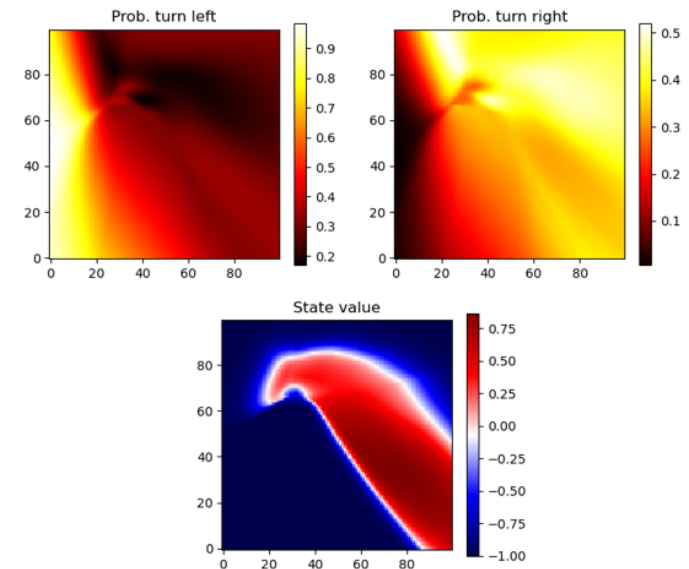DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

11

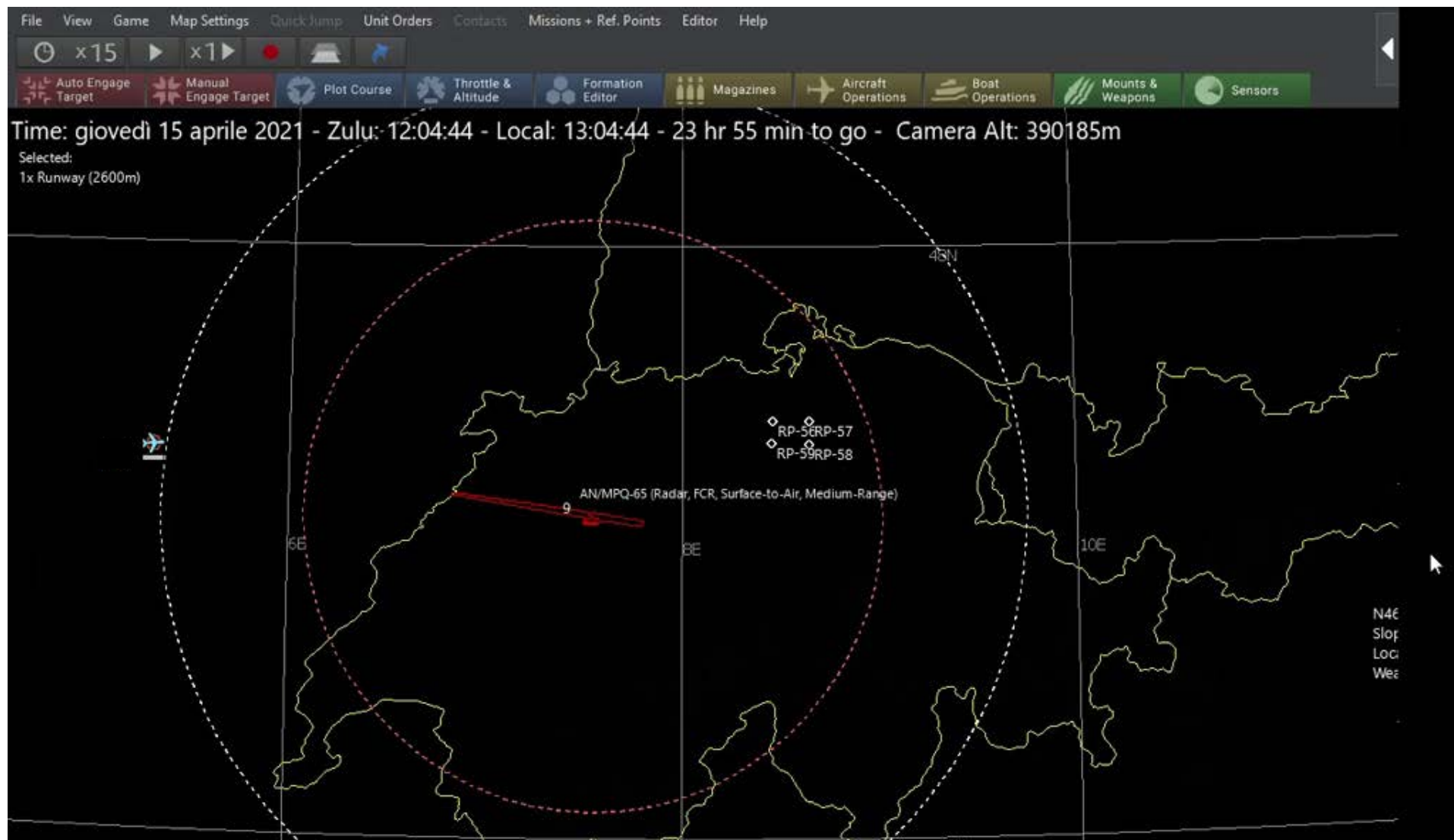MS 36369/00

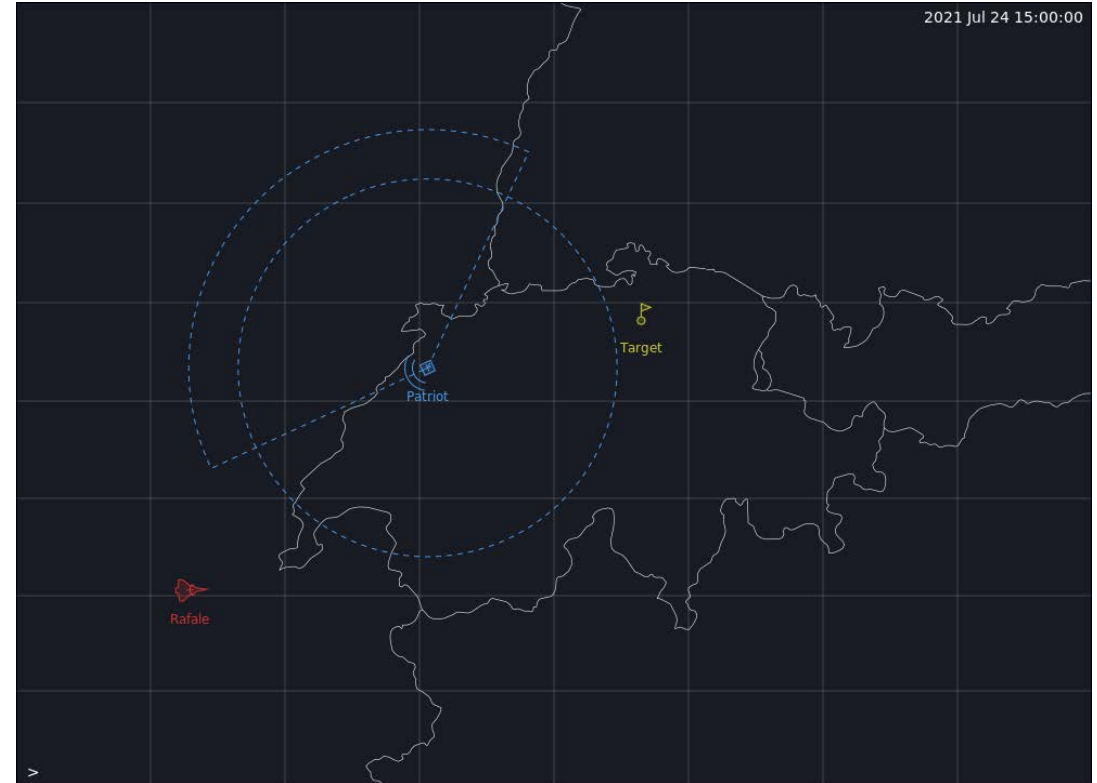# Illustration of training process of AI components:

Expert

Apprentice

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

Result: Routing of trained agent

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00

Generalization ability

Deep Self-optimizing Artificial Intelligence for Tactical Analysis, Training and Optimization
DDPS / armasuisse / S+T / Matthias Sommer, Oleg Szehr

MS 36369/00
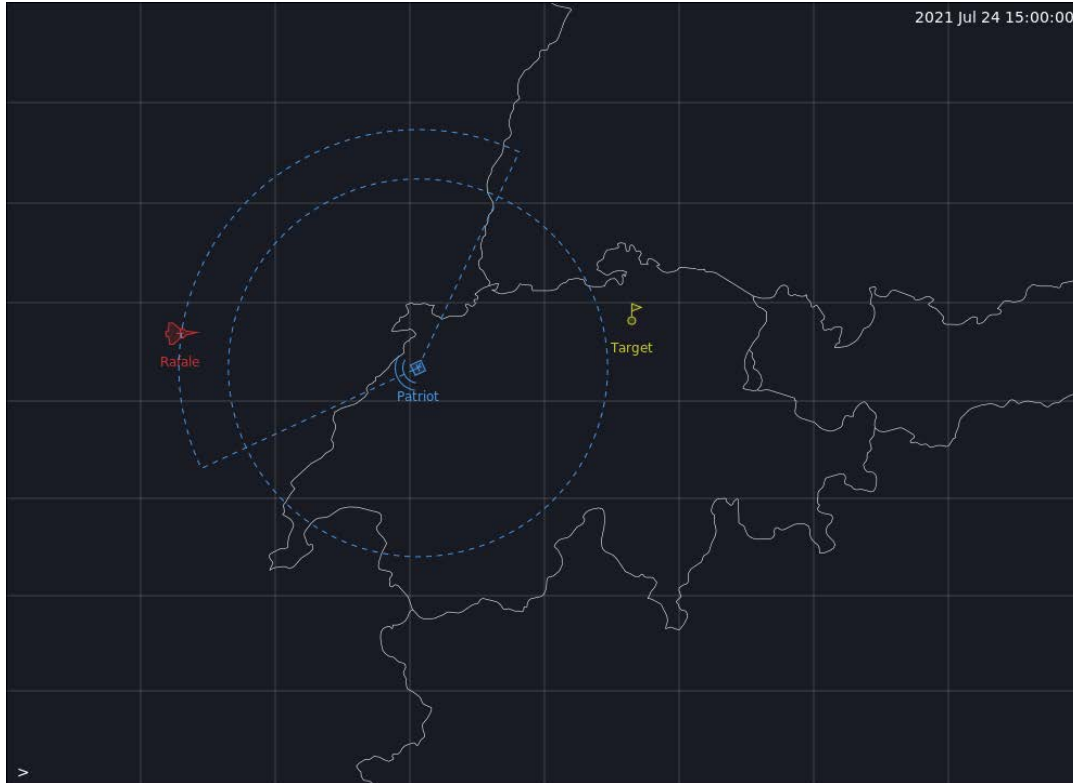
Conclusions

Proof of Concept
- Successful integration of COTS game with PyTorch
- MCTS able to learn "intelligent behavior"

Bottleneck
- Training performance (API access)
- Workaround: Training with surrogate model

Next step
- Invading aircraft can attack air defense battery